LearnDelphi.org

# Coding Boot Camp

learndelphi.org/boot-camp-2022

LearnDelphi.org
Session : Py2
Introduction to PyScripter for Python Developers
JS

# About Kiriakos Vlahos



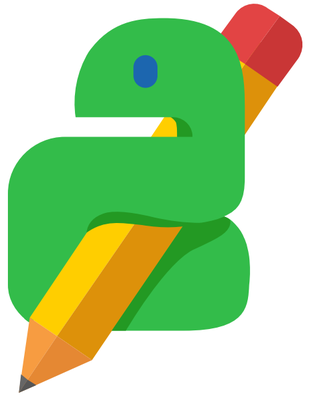
- Professor of Management Science and Data Analytics
- Open-source enthusiast
- 35 years of programming experience (Delphi, Python)
- Developer of PyScripter
- Developer/Maintainer of many Delphi libraries, notably Python for Delphi

- Github home: https://github.com/pyscripter

# Session outline



- PyScripter Overview
- Requirements and Installation
- User Interface
- The Editor
- Python Debugging



# PyScripter Overview

# PyScripter Overview



- Integrated Development Environment (IDE) for Python
- Free and open-source
- Feature-rich but lightweight
- Easy to get started, but also attractive to professionals
- Developed with Delphi
- Strong user base

# PyScripter - History



- Development started more than 15 years ago
- Initially hosted at Google Code
- Instantly became widely popular – many millions of downloads
- Development stalled when Google Code shut down
- Resumed a couple of years later at Github
- Strong following in the face of intense competition
- Recently Embarcadero provided sponsorship to the project

# Main Features

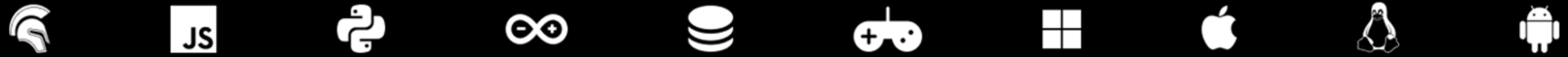


- Modern, customizable, styled user interface
- Editor with syntax highlighting and code folding
- Integrated python interpreter
- Code IntelliSence (code/parameter completion, hints, go to definition etc.)
- Integrated python debugging
- Project Management

# Advanced/Unique Features

LearnDelphi.org
Coding Boot Camp 2022

- Unit testing
- Post-mortem analysis
- Thread debugging
- Python path inspection and management
- Easy module installation with pip (external tool)
- Edit/run/debug scripts remotely in Linux and Windows servers
- Powerful integration with external tools
- Switching python versions *on the fly*.



# Requirements and Installation

# Requirements and Installation



- Windows only
- Available in 32 and 64 bits (64-bits recommended)
- Works with Python 3.2 or later (use the latest python version)
- Windows installer or registry-free zip file

# Installation Instructions



- If you have not installed python start by visiting www.python.org. Then download and install the latest python distribution. Otherwise, you can skip this step.
- Visit the Pyscripter download site and *either*:
  - download and install the Windows installer (e.g. PyScripter-4.1.1-x64-Setup.exe)
  - download the zip distribution (e.g. PyScripter-4.1.1-x64.zip) and decompress to a folder of your choice

- Note: The 64-bit version of PyScripter (x64) works only on 64-bit Windows **and** with 64-bit versions of Python. The 32-bit version (x86) of PyScripter works on both 32-bit and 64-bit Windows with 32-bit versions of Python.

# PyScripter - Links



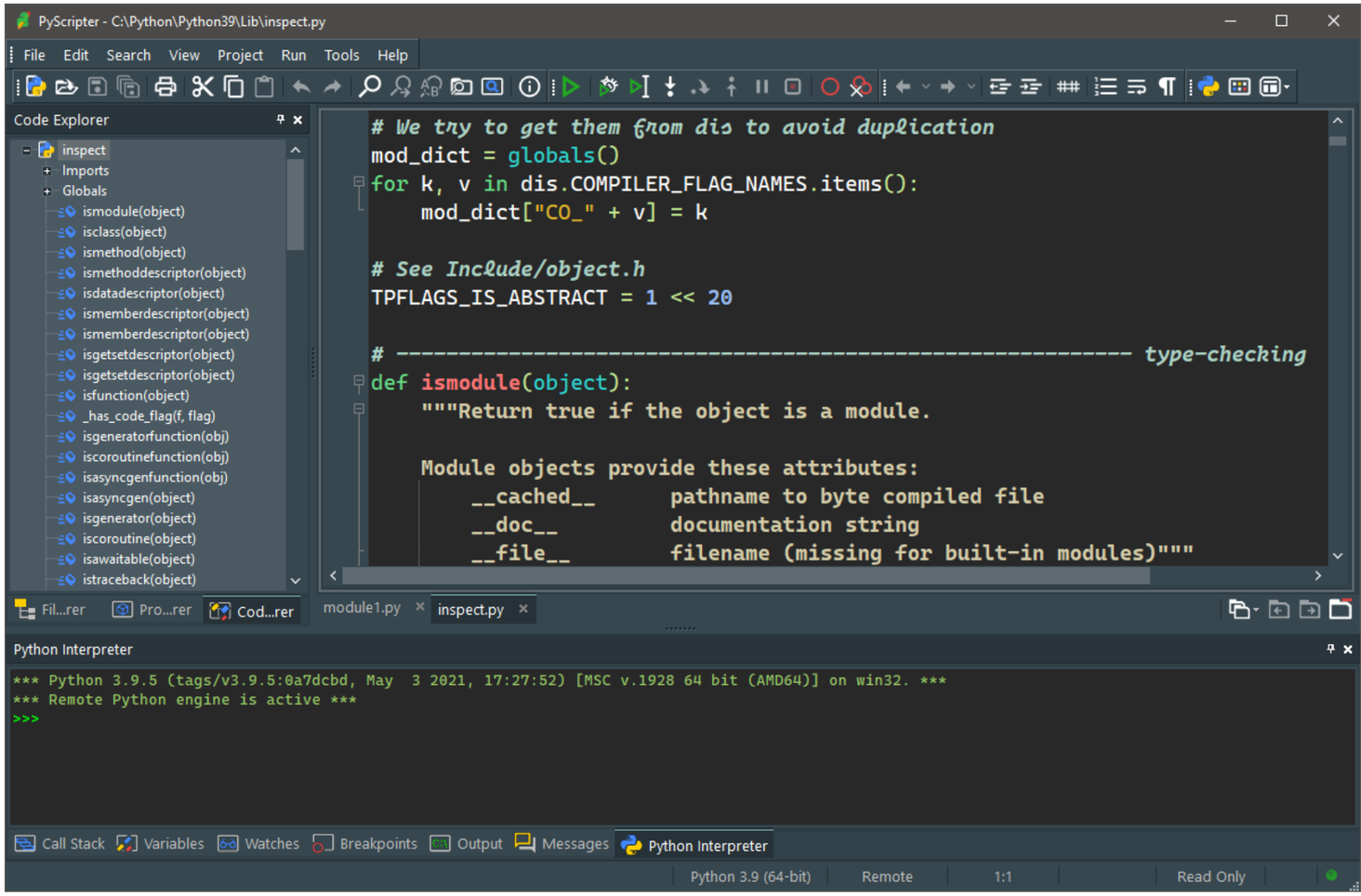
- **Home page:** https://github.com/pyscripter/pyscripter
- **Downloads**: https://sourceforge.net/projects/pyscripter/files/
- **Support:** https://groups.google.com/g/pyscripter
- **Blogs:** https://pyscripter.blogspot.com/, https://pyscripter.dev/
- **Translation home:** https://www.transifex.com/pyscripter/

User Interface

# USER INTERFACE

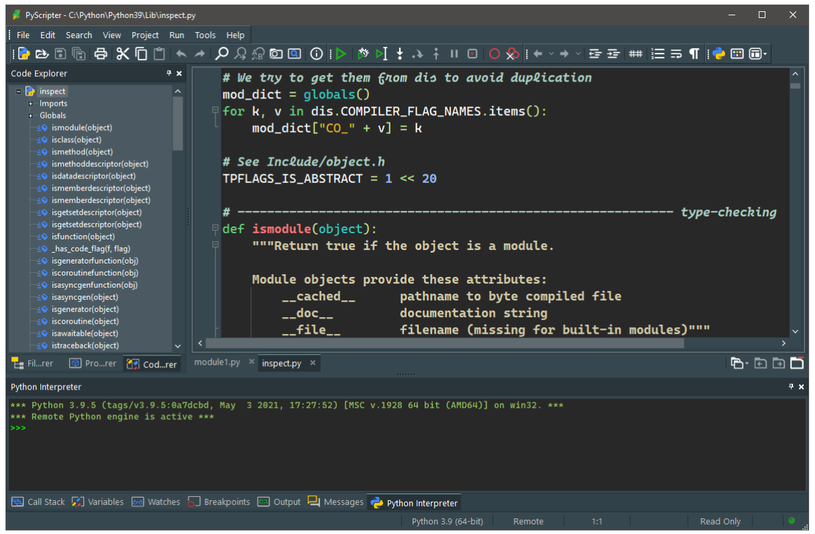


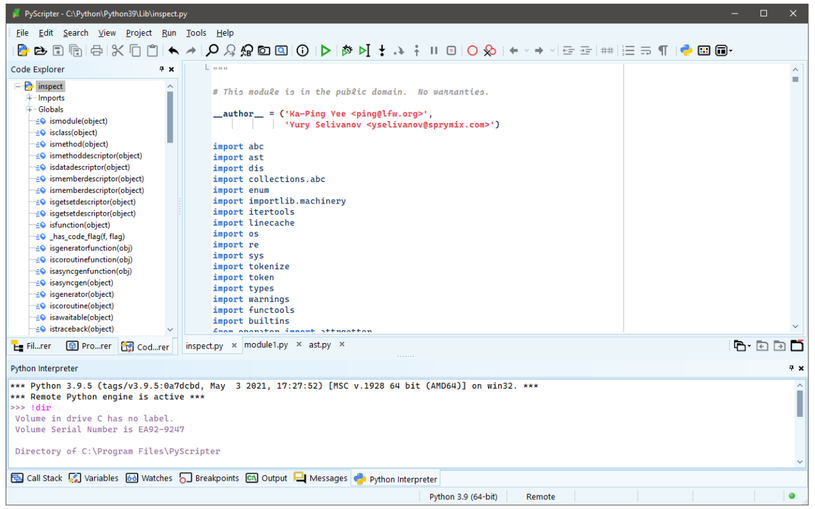
- Modern GUI with docked forms and configurable look&feel
- Editor workspace with tabs
- Material design with a bit of color
- Persistent layouts
- Switch between debug and standard layouts automatically
- UI translated in 15 languages

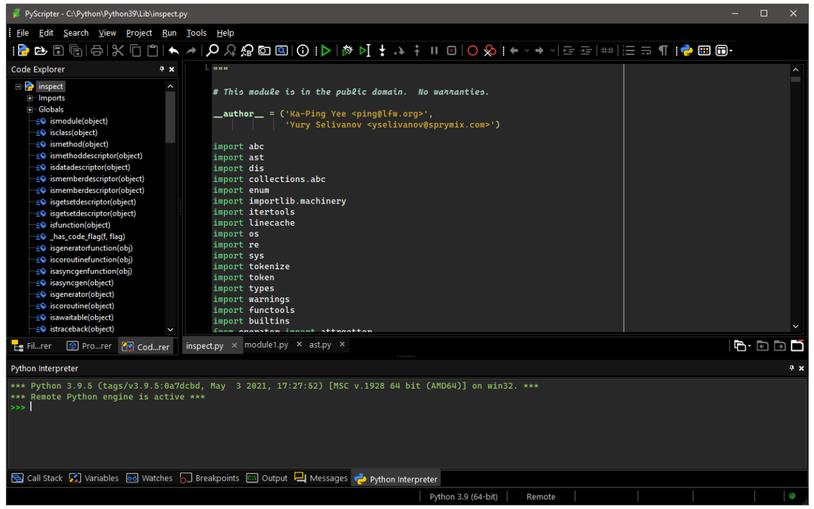


# Customizable styles and syntax highlighters

LearnDelphi.org
Coding Boot Camp 2022



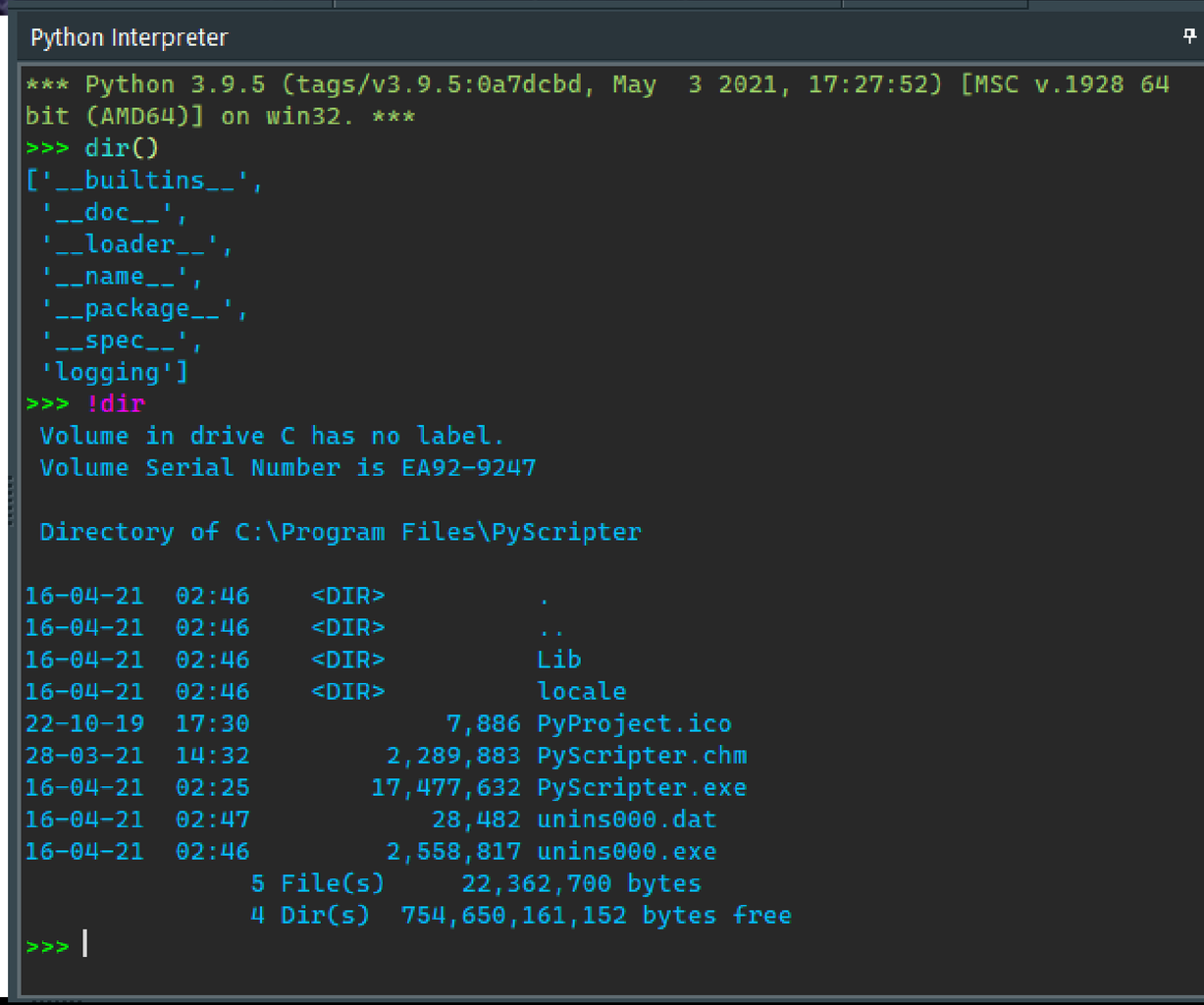



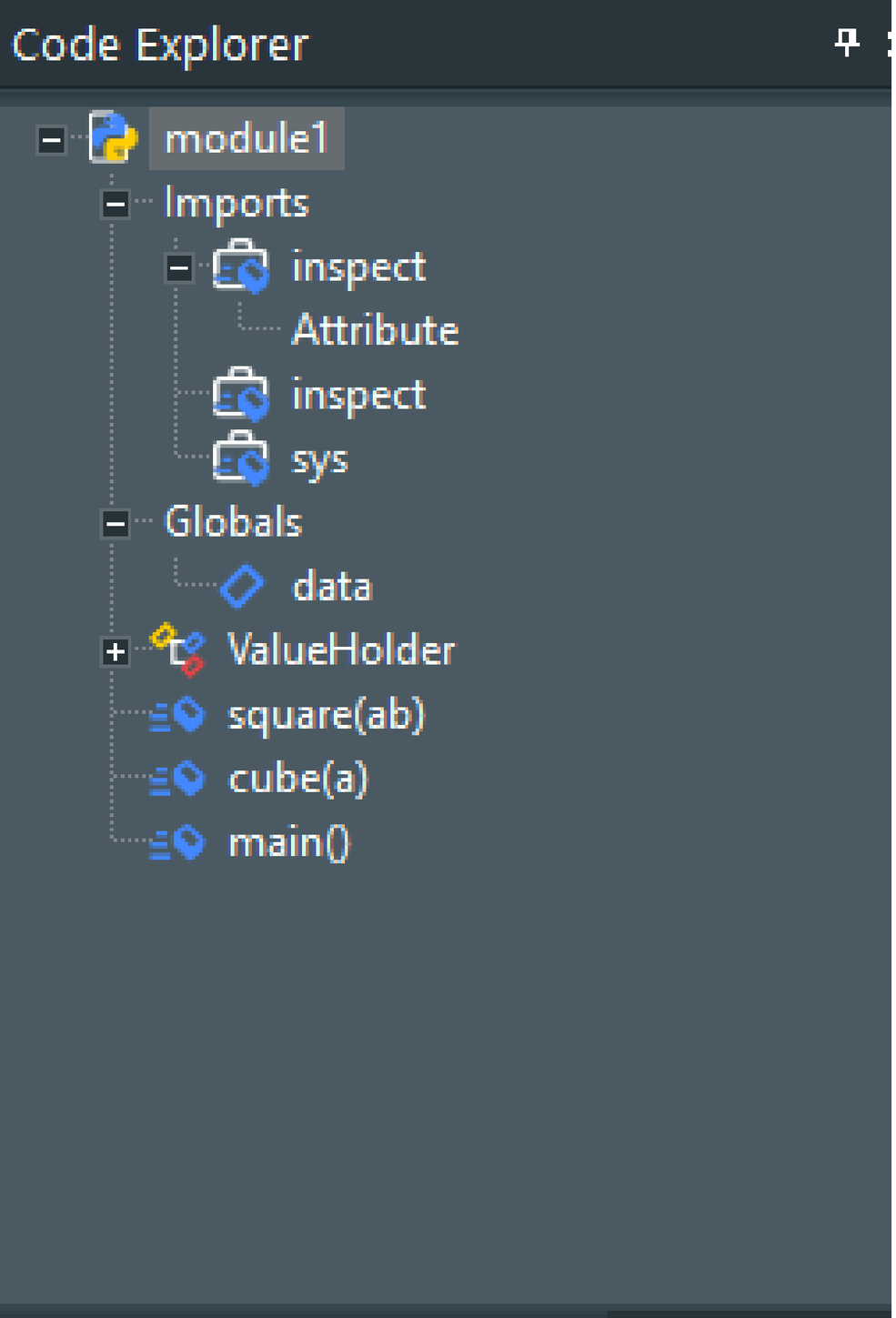



# Integrated python interpreter

LearnDelphi.org
Coding Boot Camp 2022

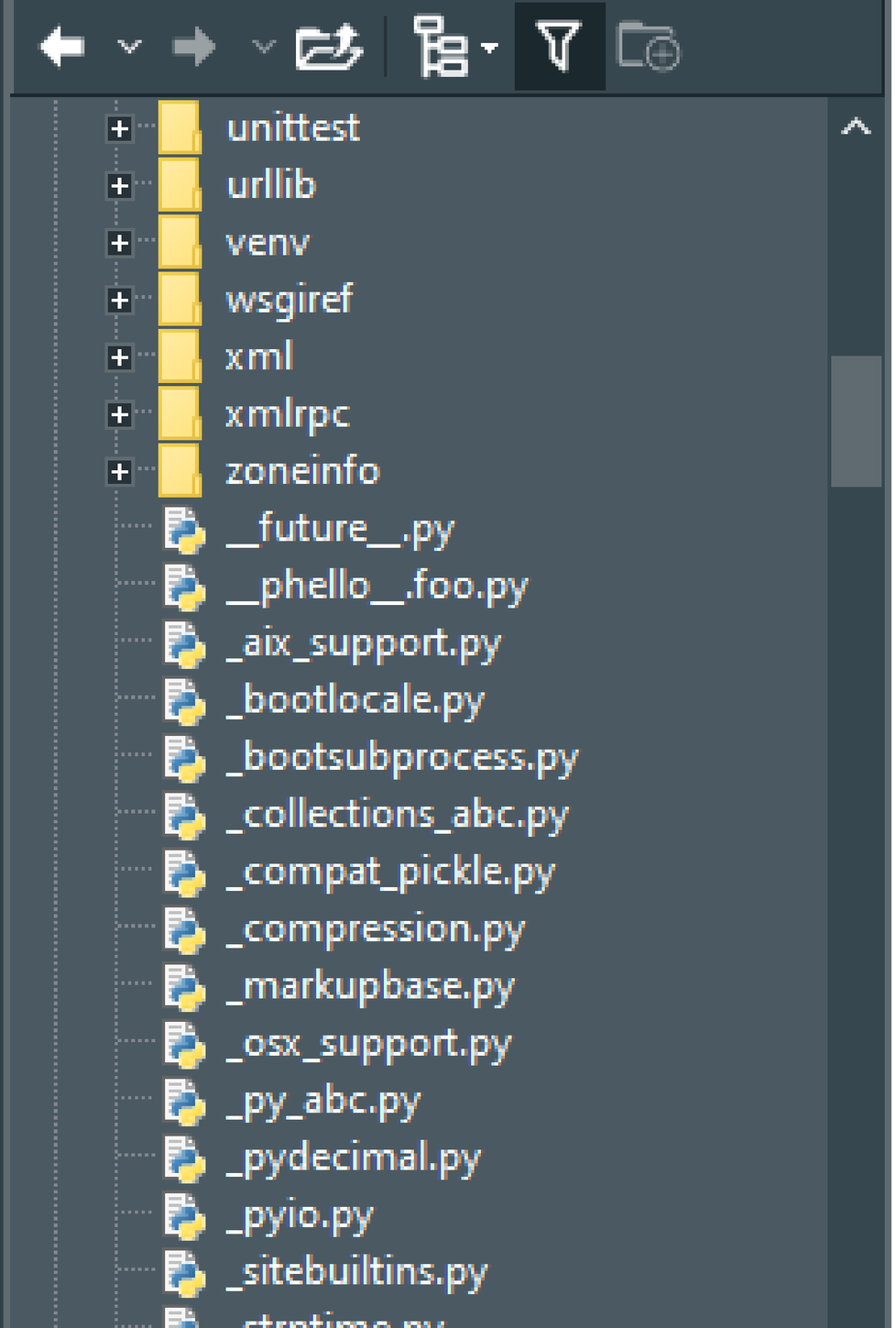
- Syntax highlighting
- Code completion
- Command history
- Serves as the *stdout* of running scripts
- Also serves as a system shell





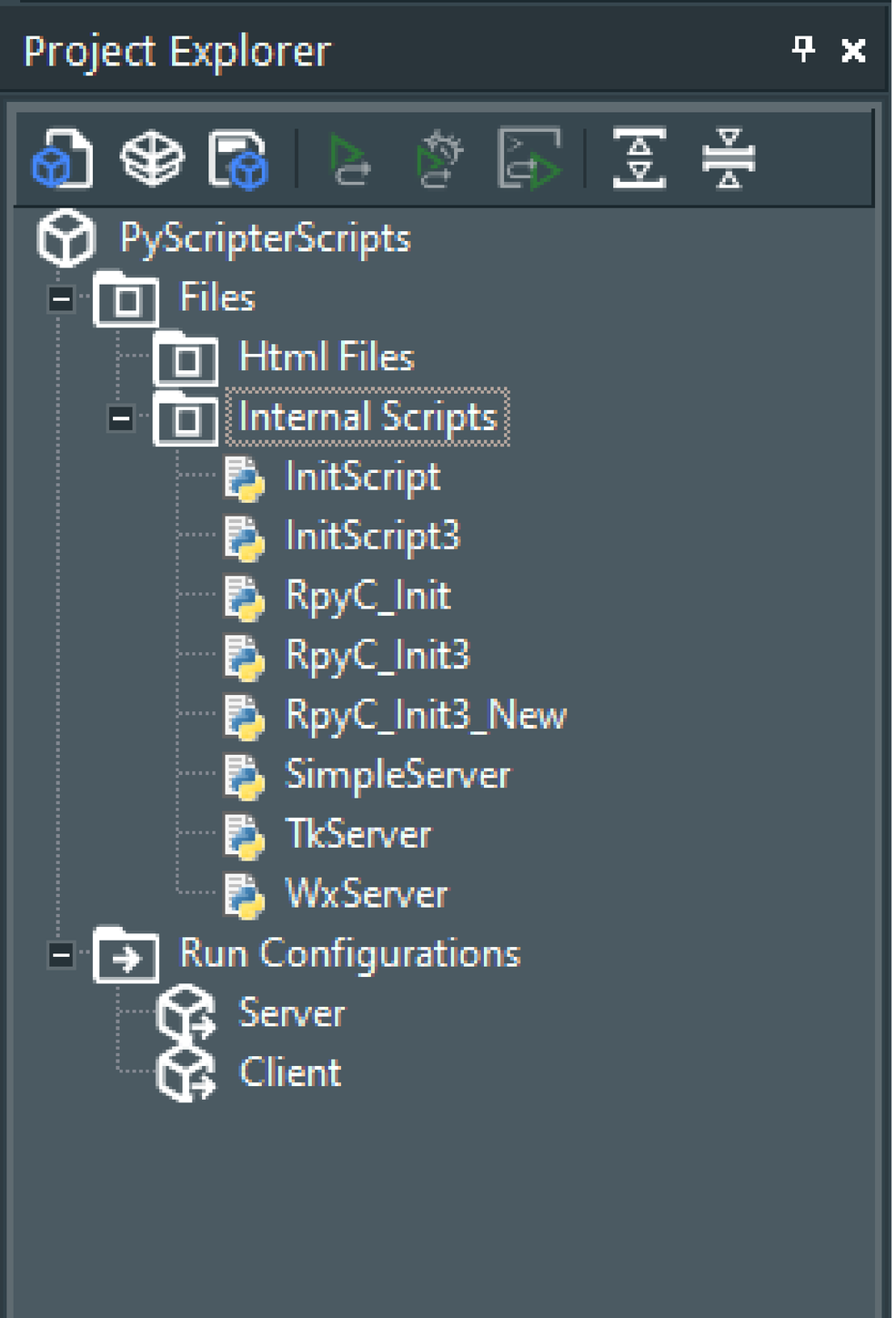

# Code explorer

- Hierarchical view of module symbols
- Can synchs position with active editor both-ways
- Can be alpha-sorted
- Hints show symbol doc comment




# File Explorer

- Windows-like File Explorer
- Same context menu as Windows
- Easy navigation to python library paths
- With Tortoise-Git can serve as a git client




# Project Explorer

- Hierarchical organization of files
- Multiple run configurations
- Easily import the contents of directories
- Project-based configuration of python path
- Search and replace project files

# Other IDE Windows



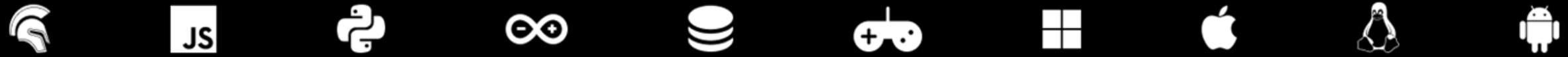
- Todo list
- Find in Files
- Regular Expression Tester
- Unit tests
- Messages Window (diagnostics)
- Output Window (output of external processes)

The Pyscripter Editor

# Python features



- Understanding Python syntax and indentation
- Run scripts without saving them
- Code IntelliSense
  - Code completion
  - Parameter completion (function signatures)
  - Code Hints
  - Go to definition (using Ctrl-click) and back-forward navigation as in Delphi
  - Find references
- Function list



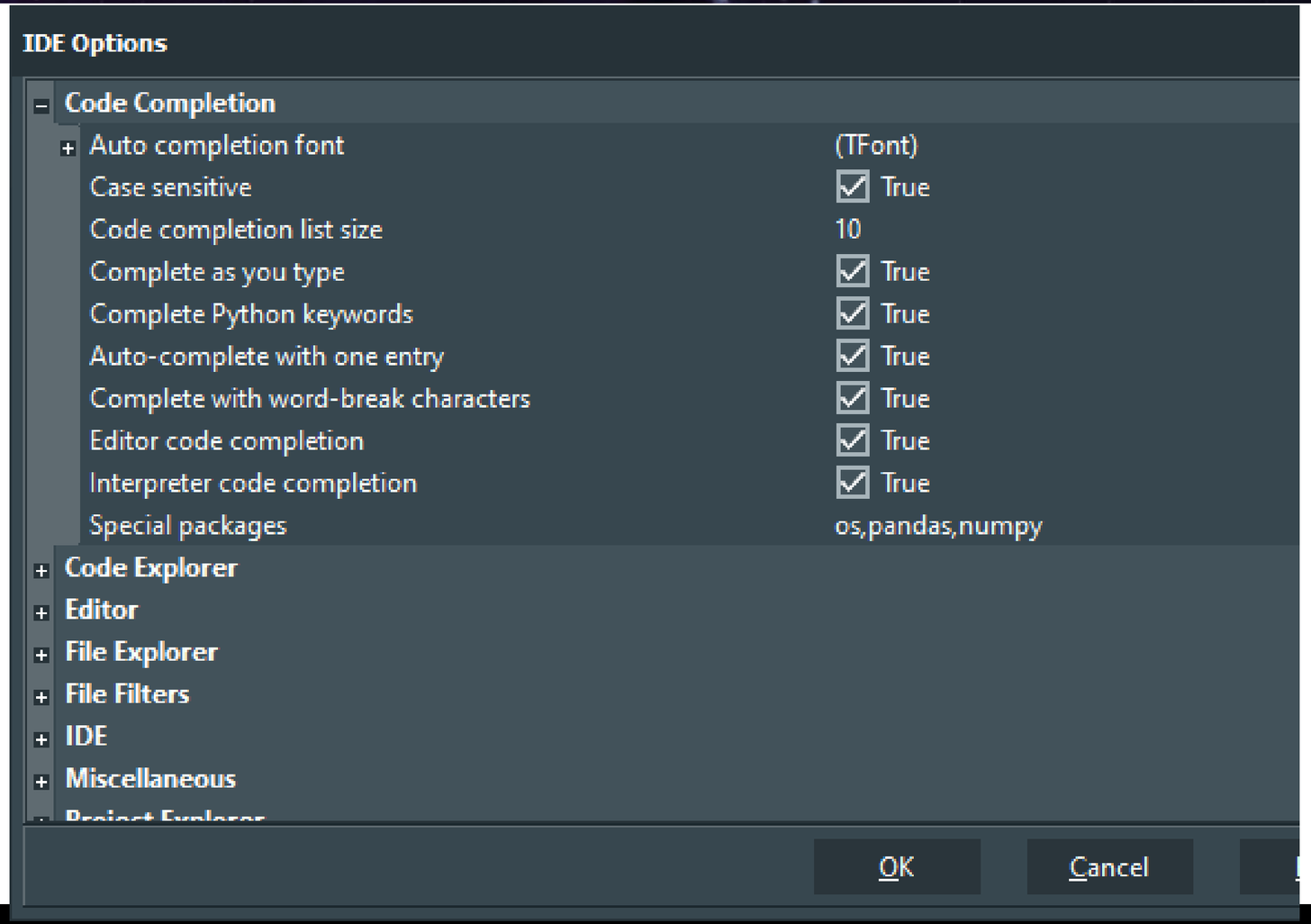
# Other editor features



- Code templates
- File templates
- Powerful parameter substitution
- OLE drag & drop supporting Unicode and html clipboard formats
- Drop target for files
- Support for font ligatures
- Several VS code commands implemented (e.g., move/copy block up/down)

# Code completion options

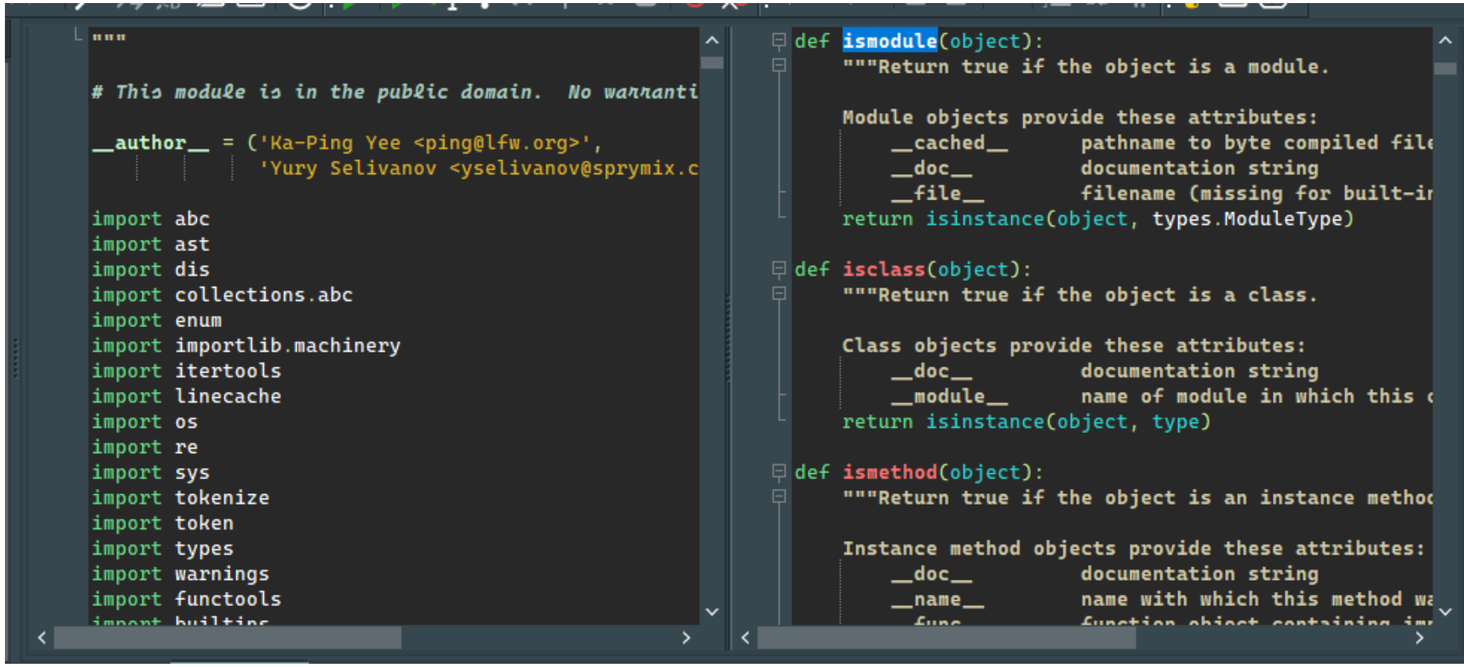




# SPLIT EDITING



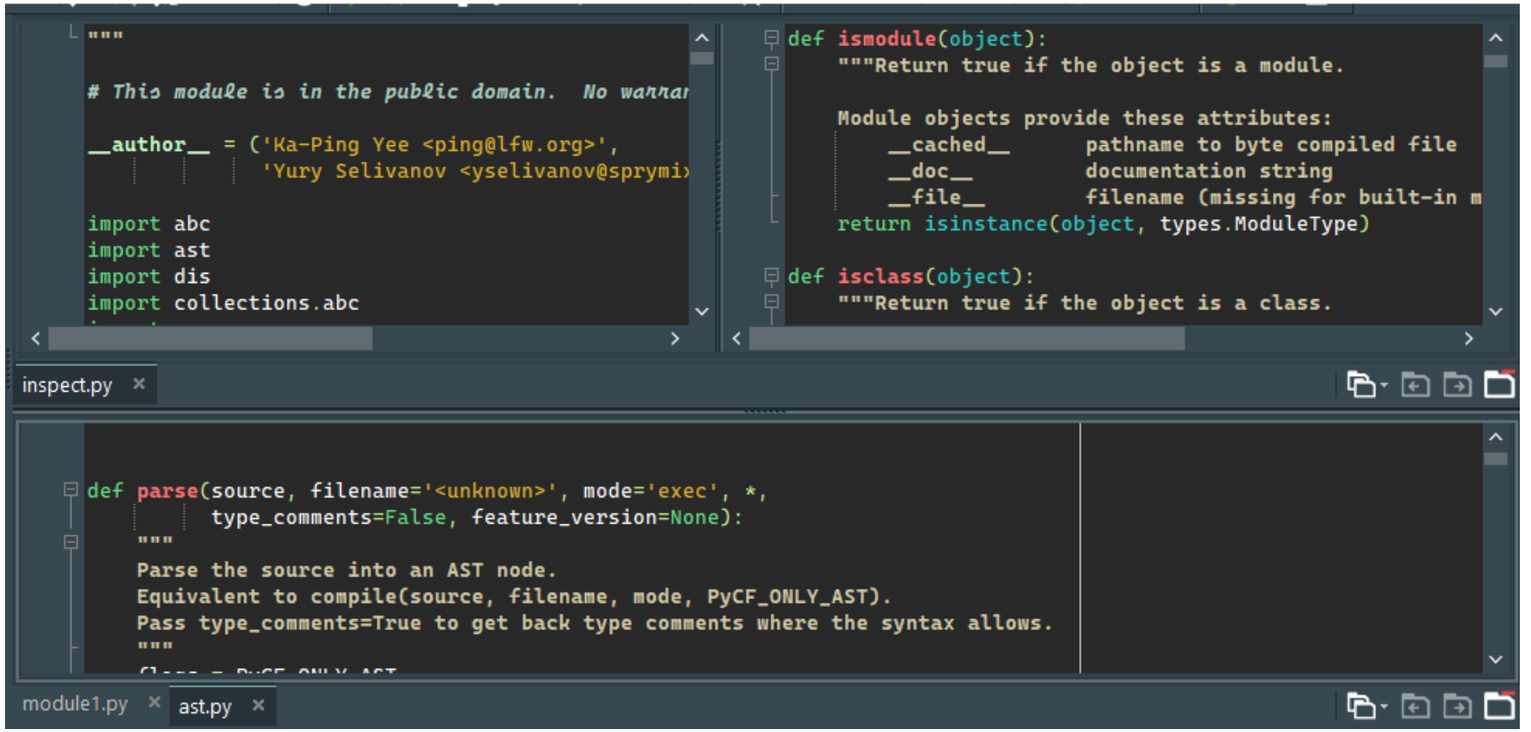
- Working on different views of the same file
- Split horizontally or vertically
- Synchronized editing



# SPLIT WORKSPACES

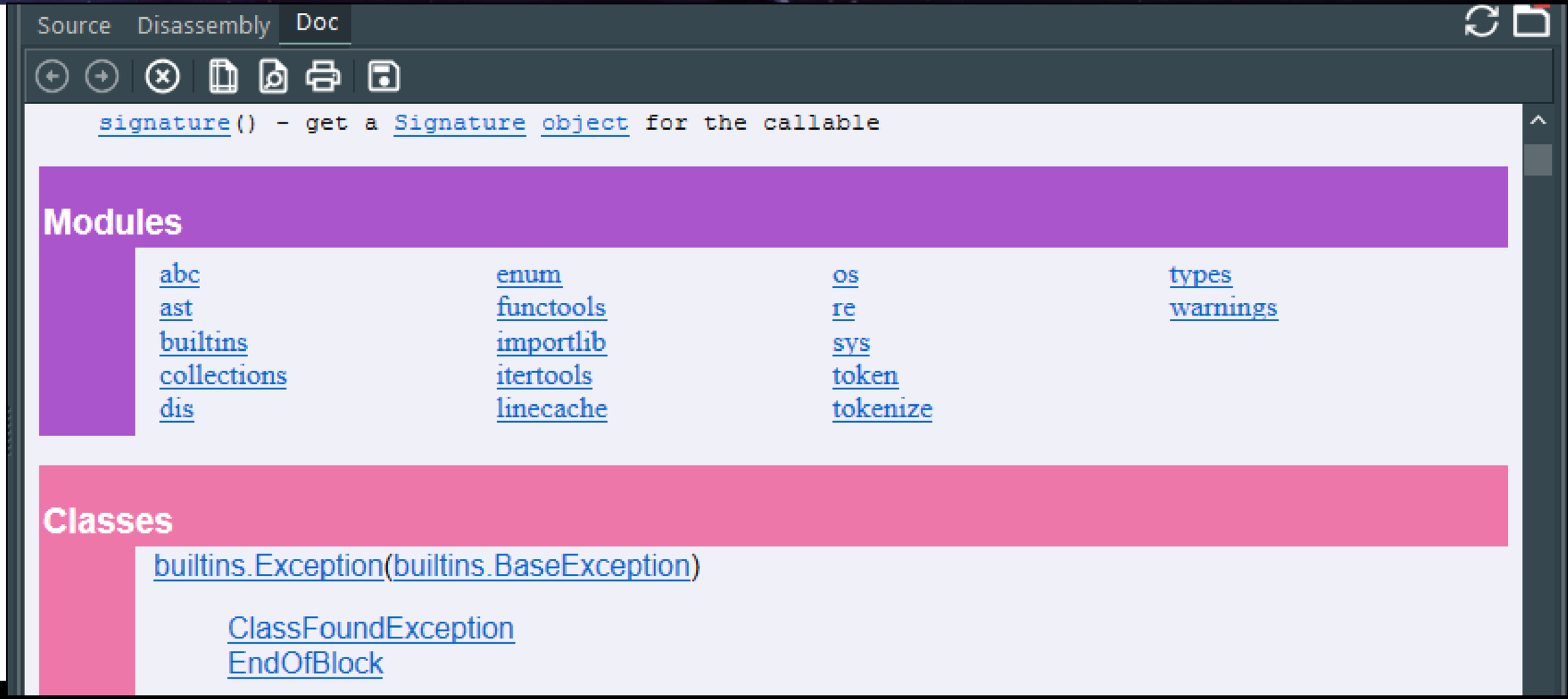


- Separate workspaces
- Split horizontally or vertically
- Move files between workspaces with drag & drop



# Editor views – Module Documentation





# Editor views - Disassembly



```
    Disassembly of inspect.py
'''

Disassembly of ArgInfo:
Disassembly of __getnewargs__:
466           0 LOAD_DEREF               0 (_tuple)
              2 LOAD_FAST                0 (self)
              4 CALL_FUNCTION            1
              6 RETURN_VALUE

Disassembly of __new__:
  1           0 LOAD_GLOBAL              0 (_tuple_new)
              2 LOAD_FAST                0 (_cls)
              4 LOAD_FAST                1 (args)
              6 LOAD_FAST                2 (varargs)
              8 LOAD_FAST                3 (keywords)
             10 LOAD_FAST                4 (locals)
             12 BUILD_TUPLE              4
             14 CALL_FUNCTION            2
             16 RETURN_VALUE

Disassembly of __repr__:
```

# Python Debugging

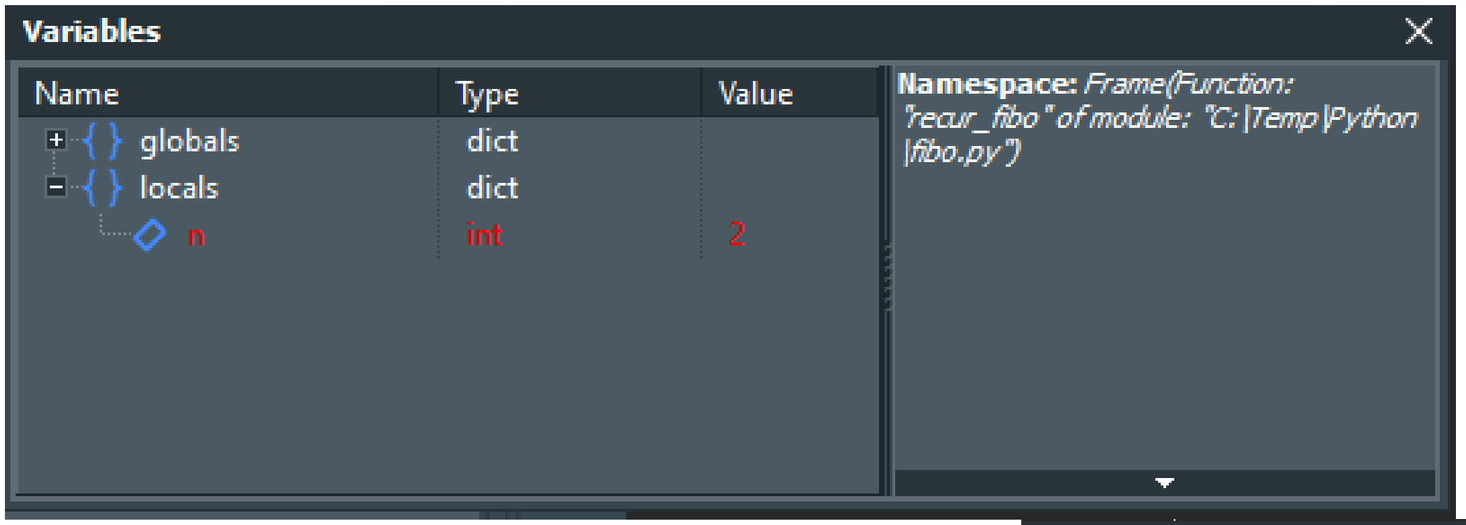
# Python debugging

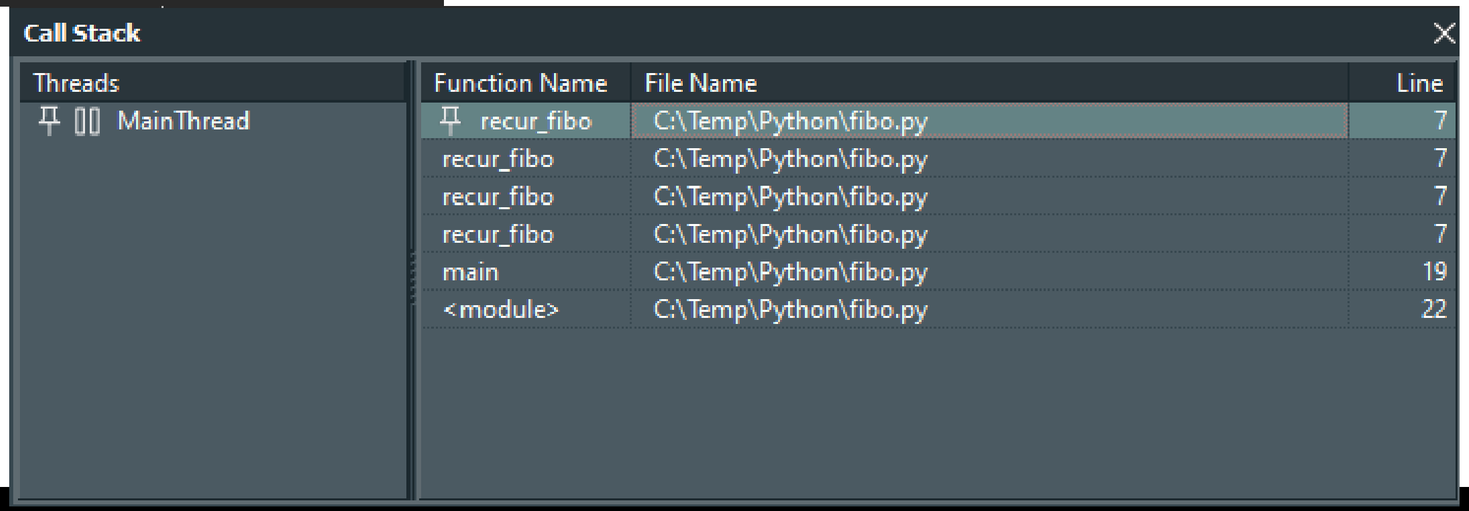


- Breakpoints
  - Enable/disable
  - Set condition
- Debugger commands
  - Debug
  - Run to cursor
  - Step into
  - Step over
  - Step out
  - Pause
  - Abort
- Debugger hints

# Debug windows I





# Debug Windows II



**Breakpoints**

| File Name | Line | Condition |
|---|---|---|
| ☐ C:\Temp\Python\BlockingThreads.py | 15 | |
| ☑ C:\Temp\Python\fibo.py | 7 | |

**Watches**

| Watches | Type | Value |
|---|---|---|
| n | int | 2 |

LearnDelphi.org

# Thank you!

Get the code:
github.com/Embarcadero/CodingBootCamp2022